ABN-192

PCI-X Gigabit Ethernet controllers

User's Guide

R0.00

Portwell Inc.

3F, No. 92, Sec. 1, Nei-Hu Rd., Taipei 114, Taiwan, R.O.C.

Headquarter: +886-2-2799-2020 FAX: +886-2-2799-1010

http://www.portwell.com.tw

About This Guide

Purpose

This manual provides information on **ABN-192** PCI-X Gigabit Ethernet card, including software installation

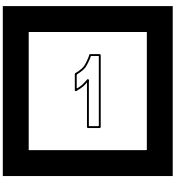
Manual Contents

Chapter 1 Overview of **ABN-192** Ethernet controllers.

Chapter 2 **ABN-192** Ethernet product detail.

Chapter 3 **ABN-192** Ethernet product programming setup detail.

Chapter 4 Driver installation detail of **ABN-192**.



Introduction

This user manual is used for ABN-192 standard universal PCI-X Ethernet controllers. The basic design of both Ethernet controller cards is identically. ABN-192 has Dual Ethernet ports and universal 64-bit standard PCI-X To enhance Ethernet performance. It designed with Intel 82546GB Gigabit Ethernet Controllers to provide two Gigabit Ethernet port. The further detail of both products will be covered in the following chapters.

Dual 64-bit Gigabit Ethernet ports base on Intel® 82546GB controller. Supports 133MHz PCI-X bus and backwards compatible with 64/32-bit Built-in Watchdog Timer (WDT) to switch Ethernet ports to bypass mode by system hang and power failure. Easy configuration of Normal / Bypass mode and WDT time-out per software programming

Built with both on-board LED for LAN status and bypass mode.

1.1 Features

- Gigabit Ethernet card with 2nd generation By-Pass function.
- Dual 64-bit Gigabit Ethernet ports base on Intel® 82546GB controller.
- 64bit 133MHz PCI-X interface.
- Built-in Watchdog Timer (WDT) function.
- Easy configuration for by-pass mode via software programming.
- Built with both on-board LED for LAN status and bypass mode,

1.2 Bypass Block Diagram

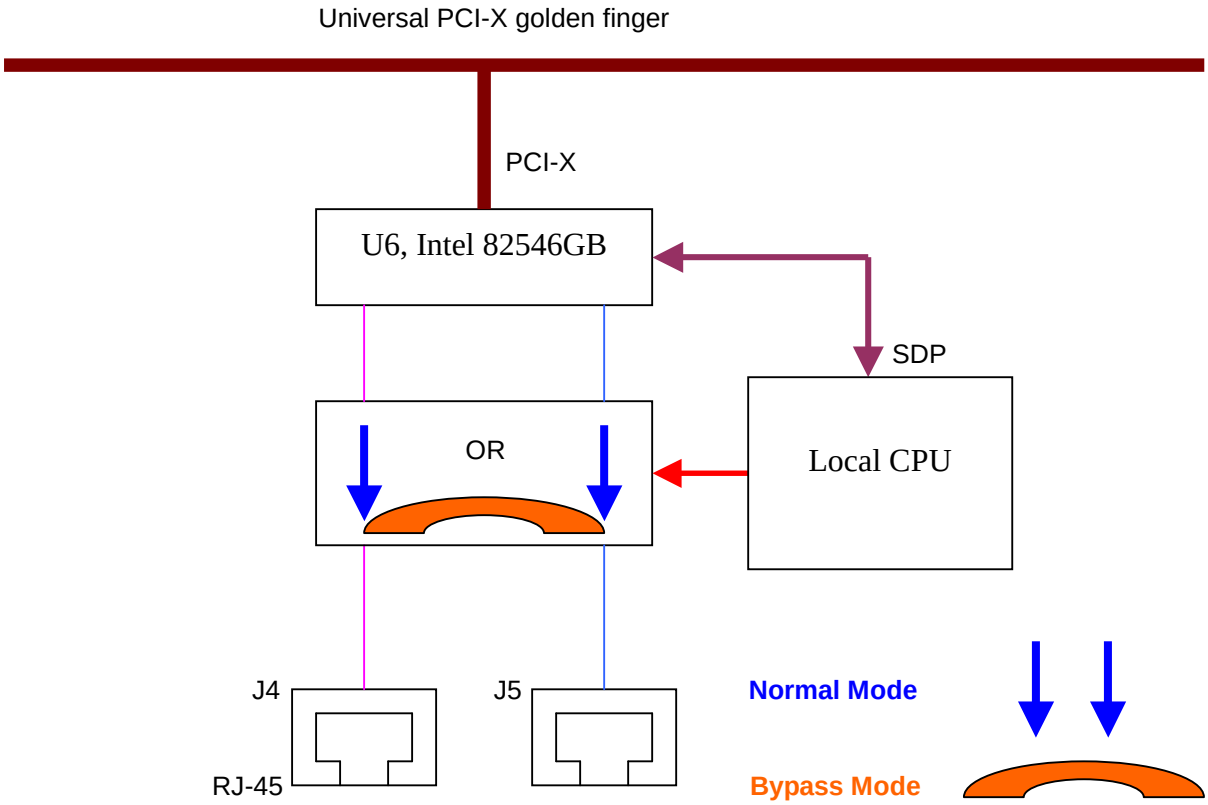


Figure 1: ABN-192 Block Diagram

2

ABN-192 PCI-X Ethernet card

2.1 Product Specification

- Gigabit Ethernet card with 2nd generation By-Pass function.
- Dual 64-bit Gigabit Ethernet ports base on Intel® 82546GB controller.
- 64bit 133MHz PCI-X interface.
- Built-in Watchdog Timer (WDT) function.
- Easy configuration for by-pass mode via software programming.
- Built with both on-board LED for LAN status and bypass mode,

2.2 Installation

Please check the included items in the box before installation. If there is any missing part, please contact the local dealer. Including items:

1. **ABN-192** PCI-X Ethernet controller card
2. Driver CD
3. User's Guide

You can follow the step to install the Ethernet controller card to the system.

- Read this Manual
- Setup the configuration from software.
- Install the card on the available PCI-X slot.
- Prepare driver for the card, which can be found in the Driver CD or download update driver form Intel website.
- Connect the Ethernet cable to check the board function.

CAUTION: Make sure the card is PROPERLY installed on a PCI-X slot; reverse or shifted installation can cause serious damage on the Ethernet controller or your system.

Programming Setup

3.1 Programming Support

The bypass test program for ABN-192 supports Linux operating system: Fedora core/Red Hat 9.0. Test programs base on gcc version 4.1.0 20060304 (Fedora Core 5). And gcc version 3.2.2 20030222(red-hat 9) The following sections will cover Linux program installation and function on OS, for other supported, please contact Portwell.

3.2 Programming Configuration

The following steps are manual setup for Linux 2.4.X/2.6.X kernel:

- Copy ABN-192 bypass program to Linux OS
- Select OS kernel version (kernel 2.4 or 2.6) and go into path.
- Go into ap path to make test program:
  [root @ host]# cd ap
  [root @ host ap]# ./complie.sh
- Run test program and go into main test screen.
  [root @ host ap]# ./test_all

Portwell ABN192 Bypass program	version:1.00
Author: Angus Yang	
function select:	
[1] set to normal mode	[2] set to bypass mode
[3] set bypass enable function	[4] set Next boot enable function
[5] watch dog timer setting	
[ESC] exit program	
Select function:1	

1. [1]set to normal mode: set bypass mode disable.
2. [2]set to bypass mode: set bypass mode enable.
3. [3]set bypass enable function: set bypass mode when the system power off

bypass enable function
function select: [1] set to bypass enable [2] set to bypass disable [ESC] return main window
Select function:1

4. [4] set next boot enable function: set bypass mode disable /enable when system boot next time.

Next boot enable fucntion
function select: [1] set next boot enable [2] set next boot disable [ESC] return main window
Select function:1

5. [5] watch dog timer setting: bypass function enabled and disable base on watch dog setting.
   - Select [1] to clear watch dog timer old setting. Reset the system to normal mode.
   - Select the watch dog timer you want to test: [4]~[e]
   - Select [3] to start watch dog timer function. The bypass mode will enabled when time is up.

watch dog timer setting
function select: [1] clear watch dog timer and set it to normal mode [2] disable bypass&watch dog timer [3] refresh bypass&watch dog timer [4] set wdt cycle:264.528 ms [5] set wdt cycle:528.516 ms [6] set wdt cycle:1057.032 ms [7] set wdt cycle:2214.065 ms [8] set wdt cycle:4228.129 ms [9] set wdt cycle:8456.258 ms [a] set wdt cycle:16912.52 ms [b] set wdt cycle:33825.03 ms [c] set wdt cycle:67550.06 ms [d] set wdt cycle:135300.1 ms [e] set wdt cycle:270600.3 ms [ESC] return main window
Select function:

*note: all functions are for single card test, it does not support more cards test at the same time now.

3.3 Programming Guide

Following is API description.

1 pci_ini

Function name	Pci_ini
Description	To find where device is
Format:	
int pci_ini(void) Input: none Return: 0: ok -1:can not find device.	

2. set_to_normal

Function name	set_to_normal
Description	Set Ethernet port to normal mode
Format:	
int set_to_normal (void) Input: none Return: 0: return ok. -1:return fail.	

3. set_to_bp

Function name	set_to_bp
Description	Set Ethernet port to bypass mode
Format:	
int set_to_bp (void) Input: none Return: 0: return ok. -1:return fail.	

4. set_nb_bpe

Function name	set_nb_bpe
Description	Set Ethernet port to bypass mode or normal mode while power start up or power off, the details as below description
Format:	
int set_nb_bpe (unsigned char para) Input: para SET_NB_BPE_NB0: set Ethernet port to normal mode when power start up. SET_NB_BPE_NB1: set Ethernet port to bypass mode when power start up. SET_NB_BPE_BPE0: set Ethernet port to normal mode when power off. SET_NB_BPE_BPE1: set Ethernet port to bypass mode when power off. Return: 0: return ok. -1:return fail.	

5 set_period_wdt

Function name	set_period_wdt
Description	Set watch dog timer period
Format:	
int set_period_wdt(unsigned char pd) Input: pd SET_WDT_CYCLE_264258:set watch dog timer period as 264.258ms SET_WDT_CYCLE_528516:set watch dog timer period as 528.516ms SET_WDT_CYCLE_1057032:set watch dog timer period as 1057.032ms SET_WDT_CYCLE_2114065:set watch dog timer period as 2114.065ms SET_WDT_CYCLE_4228129:set watch dog timer period as 4228.129ms SET_WDT_CYCLE_8456258:set watch dog timer period as 8456.258ms SET_WDT_CYCLE_1691252:set watch dog timer period as 16912.52ms SET_WDT_CYCLE_3382503:set watch dog timer period as 33825.03ms SET_WDT_CYCLE_6755006:set watch dog timer period as 67550.06ms SET_WDT_CYCLE_1353001:set watch dog timer period as 135300.1ms SET_WDT_CYCLE_2706003:set watch dog timer period as 270600.3ms Return: 0: return ok. -1:return fail.	

6. refresh_bp_wd

Function name	refresh_bp_wd
Description	Refresh Ethernet to watch dog enable and bypass mode.
Format:	
Modify it to eeprom int refresh_bp_wdt(void) Input: none Return: 0: return ok. -1:return fail.	

7. clr_wdt_to_sts

Function name	clr_wdt_to_sts
Description	Clear the set of watch dog and go into normal mode.
Format:	
1. Modify it in eeprom. int clr_wdt_to_sts(void) Input: none Return: 0: return ok. -1:return fail.	

8. dis_bp_wdt

Function name	dis_bp_wdt
Description	Disable watch dog timer and keep status as current state
Format:	
1. Modify its setting in eeprom 2. If Ethernet is on bypass mode, when user does this command, it will be disable watch dog and keep it in bypass mode. If normal mode, after doing this command, it will be kept in normal mode. int dis_bp_wdt (void) Input: none Return: 0: return ok. -1:return fail.	

9 read_bp_status

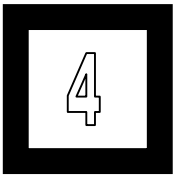
Function name	read_bp_status
Description	Read bypass status
Format:	
unsigned char read_bp_status (void) Input: none Return: Bit0=0: normal mode Bit0=1: bypass mode	

10 read_bp_wdt_status

Function name	read_bp_wdt_status
Description	Read bypass and watch dog status
Format:	
unsigned char read_bp_wdt_status (void) Input: none Return: Bit1=0: watch dog disable. Bit1=1: watch dog enable. Bit0=0: watch dog time out not occur Bit0=1: watch dog time out	

11 read_nb_bpe_status

Function name	read_nb_bpe_status
Description	Read bypass and watch dog status
Format:	
unsigned char read_nb_bpe_status (void) Input: none Return: Bit1=0: next boot to 0. Bit1=1: next boot to 1. Bit0=0: bypass enable to 0 Bit0=1: bypass enable to 1	



Driver Installation

4.1 Driver Support

The drivers for **ABN-192** support the following OS Windows 2000, XP, NT, 98/98SE, ME, 95, DOS, Novell Netware 4.11/higher and Linux Kernel 2.2.x/2.4.x. The following sections will cover driver installation on some OS, for other supported drivers, please contact Portwell or refer to Intel

4.2 Driver Setup for Windows® 2000

The following steps are manual installation for Windows 2000

- Boot Windows 2000 then insert Driver CD to CD-ROM.
- Double-click pro2kxp.exe in ***\Pro1000\Win\Win2KXP***, and then An Wizard window will pop up click ***Next>***.
- Click ***NEXT>*** button, the Wizard summary window will appear.
- Click ***Finish*** button, then click ***CLOSE*** button.

4.3 Driver Setup for Windows® NT

The following steps are manual installation for Windows NT

- Boot Windows NT then insert Driver CD to CD-ROM.
- Double-click pront4.exe in ***\Pro1000\Win\Winnt4*** , and then An Wizard window will pop up click ***Next>***.
- Click ***NEXT>*** button, the Wizard summary window will appear.
- Click ***Finish*** button, then click ***CLOSE*** button.
- Reboot the system.

4.4 Driver Installation for Linux

Some of the kernel has the driver for ABN-192, and it will automatically add the new hardware when booting up. If the driver is not included in the kernel, please following the step to install.

- Get the driver from CD or download from Intel website.

NOTE: For ABN-192, the driver is **e1000-Driver_Version.tar.gz**

- Make the file and install to the machine

  Example: Installation of e1000-7.0.33.tar.gz on Fedora core 4.

  ```
  [root @ host~]# tar zxvpf e1000-7.0.33.tar.gz
  [root @ host~]# cd e1000-7.0.33/src/
  [root @ host src]# make install
  [root @ host src]# modprobe e1000
  ```

NOTE: Please consult with your vender or read the reference document about compiling driver for different version of kernel.

- Reboot the system if necessary.